A multi-agent system for enabling collaborative situation awareness via position-based stigmergy and neuro-fuzzy learning

Giovanna Castellano [a], Mario G.C.A. Cimino [b,*], Anna Maria Fanelli [a], Beatrice Lazzerini [b], Francesco Marcelloni [b], Maria Alessandra Torsello [a]

[a] Department of Computer Science, University of Bari "A. Moro", Via E. Orabona 4, Bari 70126, Italy
[b] Department of Information Engineering, University of Pisa, Largo Lucio Lazzarino 1, Pisa 56122, Italy

A B S T R A C T

Situation awareness is a computing paradigm which allows applications to sense parameters in the environment, comprehend their meaning and project their status in the next future. In collaborative situation awareness, a challenging area in the field of Ambient Intelligence applications, situation patterns emerge from users′ collective behavior. In this paper we introduce a multi-agent system that exploits positioning information coming from mobile devices to detect the occurrence of user′s situations related to social events. In the functional view of the system, the first level of information processing is managed by marking agents which leave marks in the environment in correspondence to the users′ positions. The accumulation of marks enables a stigmergic cooperation mechanism, generating short-term memory structures in the local environment. Information provided by such structures is granulated by event agents which associate a certainty degree with each event. Finally, an inference level, managed by situation agents, deduces user situations from the underlying events by exploiting fuzzy rules whose parameters are generated automatically by a neuro-fuzzy approach. Fuzziness allows the system to cope with the uncertainty of the events. In the architectural view of the system, we adopt semantic web standards to guarantee structural interoperability in an open application environment. The system has been tested on different real-world scenarios to show the effectiveness of the proposed approach.

Keywords:
Collaborative situation awareness
Multi-agent system
Emergent paradigm
Fuzzy information granules
Neuro-fuzzy model

1. Introduction

Situation awareness is a computing paradigm that enables applications to sense and explore situations in which the users are, with the aim of predicting their demands at a certain time [1]. The paradigm relies on the *context*, that is, all the relevant data and information (e.g., the user′s position in space and time, the surrounding things and events) which can help comprehending what is happening in the environment [2–4]. This form of autonomous perception implies reasoning, decision, adaptation, and other characters of cognitive systems [5], as well as dealing with an intrinsic uncertainty in data [6,7].

To this aim, Korpipää et al. [8] have proposed a framework for managing uncertainty in raw data and inferring higher-level context abstractions with a related probability. Fuzzy sets are employed to convert unstructured raw data into a representation defined in a context ontology through predefined fuzzy labels. Situations are recognized by means of a basic Bayes classifier, which learns conditional probabilities from training data for each situation. In [9] fuzzy quantization is used to convert raw sensor data into context information. Such information is exploited by fuzzy controllers for adapting applications to the specific context. However, no semantic description of context is considered. Ranganathan et al. [10] have modeled uncertainty in situation awareness by associating a confidence value with all pieces of contextual information. The authors adopt three methods to infer the user′s situation: (i) probabilistic logic, (ii) fuzzy logic, and (iii) Bayesian networks.

In [11] uncertainty is managed by first extending the context ontology so as to allow additional probabilistic markups and then by adopting Bayesian networks to infer the current situation of the user. In [12] contextual information is codified in the antecedent part of linguistic rules whose consequent parts express the degree of confidence in the occurrence of a situation. Weights can be specified to represent the relative importance of each contextual condition for inferring a situation. In [13] a neuro-fuzzy classification system is trained to map sets of contextual information to particular situations by fuzzy rules.

In [7,14] we have proposed a design method for managing situation awareness. This method is based on the concurrent use

* Corresponding author. Tel.: +39 050 2217 455; fax +39 050 2217 600.
E-mail addresses: castellano@di.uniba.it (G. Castellano), m.cimino@iet.unipi.it (M.G.C.A. Cimino), fanelli@di.uniba.it (A.M. Fanelli), b.lazzerini@iet.unipi.it (B. Lazzerini), f.marcelloni@iet.unipi.it (F. Marcelloni), torsello@di.uniba.it (M.A. Torsello).

of a semantic and a fuzzy engine. The semantic engine can infer one or more situations exploiting symbolic information. When multiple situations are inferred, a fuzzy engine computes a certainty degree for each situation, taking the intrinsic vagueness of some conditions of the semantic rules into account.

The structure of rules has been designed according to an upper situation ontology which is domain independent. The user calendar acts as a reference for the parameterization of such fuzzy rules for each user. The use of a calendar is however an *explicit* input required to the user. On the contrary, context information should be collected in terms of collaborative *implicit* input, coming from changes in the environment.

To avoid using explicit inputs as context sources, in [15,16] we have proposed an approach based on the *emergent* paradigm [5] for automatically detecting social events (e.g., meetings, conferences, festivals, entertainment, and so on) by exploiting a position-based stigmergy paradigm. Stigmergy can be defined as an indirect communication mechanism that allows simple entities to structure their activities through the local *environment* [17]. The approach has been referred to as collaborative situation awareness (CSA). In particular, each user is associated with one marking agent which leaves periodically marks in the environment in correspondence to his position. In a stigmergic computing scheme, the environment acts as a common shared service for all entities enabling a robust and self-coordinating mechanism. The accumulation of these marks is monitored by an event agent which detects events based on a fuzzy information granulation process. Finally, situation agents infer user situations from the underlying events. The inference process is performed by fuzzy rules generated by an expert taking some mathematical constraints into consideration.

In this paper, we extend our approach by focusing on a multi-agent architecture. Further, in order to make the approach completely independent of the user's inputs, we generate the fuzzy rules by exploiting a neuro-fuzzy system. We adopt Gaussian membership functions and train the neuro-fuzzy system by tracing a number of users involved in a social event. We need only to know the number of users who participate in the event. The proposed scheme is tested on four representative real scenarios, considering four different types of situation. For each scenario, the scheme has proved to be able to recognize the four types of situation just approximately at the instants when these situations occur.

The paper is organized as follows. In Section 2, we introduce the functional view of the system. Section 3 shows the architecture of the system, by focusing on the knowledge representation. In Section 4, we discuss some experimental results. Section 5 draws some final conclusion.

2. The functional view of the system

Situation awareness is achieved in our multi-agent system by exploiting three processing levels: the marking, the fuzzy granulation and the inference processing levels. In this section, we will describe how the three levels work and interact with each other. The first two levels will be discussed shortly. The interested reader can refer to our previous paper [16] for details. The third level will be analyzed in depth. Indeed, unlike in [16], where fuzzy partitions were generated heuristically, here we adopt a neuro-fuzzy approach. Further, we employ slightly different fuzzy rules which determine the situation at a certain instant by considering the certainty degrees of the situations at the previous time step.

2.1. The marking processing level

We consider the spatial area under observation normalized in $[0,1]\times[0,1]$ and superimpose on this area a grid consisting of L^2 squares, where each square Q is identified by a pair of coordinates (x,y), with $x,y\in[1,\dots,L]$. The size of the area and the number of squares depend on the specific application domain. Each user is associated with a *Marking Agent* (MA), which periodically leaves a mark at the position where the user is currently located. Each mark is specific to an MA and is characterized by an intensity with a spatial and a temporal decay. In particular, the intensity decreases with the increase of the distance from the position of the user and with the passing of the time. The time period of the intensity decay is longer than the time period used by the MAs for leaving marks. Thus, if the user is still in a specific position, new marks at the end of each period will superimpose on the old marks and the intensity will reach a stationary level. On the contrary, if the MA moves to other locations, the mark intensities will decrease with the passage of the time without being reinforced.

Formally, at each instant $\bar{t}$, $\bar{t}=0,T_M,2T_M,\dots$, the MA_i leaves in the squares $Q(x,y), x,y\in[1,\dots,L]$, a mark of intensity $I_i(x,y,\bar{t})$ defined as

$$I_{i,\bar{t}}(x,y,\bar{t})=\max(0,I_{MAX}\cdot[1-\delta\cdot\max(|x-x_p|,|y-y_p|)]) \quad (1)$$

Every T_D seconds the intensity of the mark decays of a percentage α of its current value, that is,

$$I_{i,\bar{t}}(x,y,t)=\alpha\cdot I_{i,\bar{t}}(x,y,t-T_D) \quad (2)$$

with $t=\bar{t}+T_D,\bar{t}+2T_D,\dots$

For each square $Q(x,y)$, the actual value $I(x,y,t)$ of the intensity is obtained as the sum of the intensities of the marks left by each MA, that is,

$$I(x,y,t)=\sum_{\forall i,\forall\bar{t}:I_{i,\bar{t}}(x,y,t)>0} I_{i,\bar{t}}(x,y,t) \quad (3)$$

The intensities of the marks are granulated in the second processing level by two event agents (EAs), namely the Grouping Agent (GA) and the Disjoining Agent (DA). Both the GA and the DA agents are generated by an MA whenever the mark left by the MA itself is superimposed on at least one mark left by other MAs.

The control logic of a generic marking agent MA_i can be summarized as follows:

Loop
Wait for T_M seconds;
Leave a mark of Intensity $I_i(x,y)$ in the squares $Q(x,y)$;
Ask the Environment whether, in at least one square with $I_i(x,y)>0$, there exists another mark left by another MA_j with intensity $I_j(x,y)>0$;
If there exists such mark
Then Generate a *GA* and a *DA*;
End loop

2.2. The fuzzy granulation processing level

The GA characterizes the behavior of groups of MAs and is devoted to detect when a grouping event occurs. Once instantiated, each GA observes a neighboring area, here denoted by $N(x_G,y_G)$, centered in the position (x_G,y_G) of the GA. The position (x_G,y_G) coincides with the position (x_P,y_P) of the MA which generates the GA. As a consequence, the GA follows the same movements as the corresponding MA. We assume that the size of the area $N(x_G,y_G)$ is equal to the size of the area of a mark. The intensity associated with the area $N(x_G,y_G)$ is computed as

$$I_{GA}(x_G,y_G,t)=\sum_{(x,y)\in N(x_G,y_G)} I(x,y,t) \quad (4)$$

GAs corresponding to the same group of users are fused in such a way that only one GA is associated with a group of users. Two

GAs are fused when at least one square of the neighborhood of the former is superimposed on one square of the neighborhood of the latter. At each instant $\bar{t}$, the position (x_G, y_G) of the generated GA is computed as the center of gravity of the positions (x_P, y_P) of the MAs which have instantiated the fused GAs.

The DA detects if a user, after having joined a group, separates from it. The position (x_D, y_D) of a DA coincides at each time step with the position of the corresponding MA. A DA is removed when the GA, which contains the user corresponding to the DA, is removed. Once instantiated, each DA observes a neighboring area, here denoted by $N(x_D, y_D)$, centered in (x_D, y_D). We assume that also the size of $N(x_D, y_D)$ is equal to the size of the area of a mark. The intensity associated with the area $N(x_D, y_D)$ is computed as

$$I_{DA}(x_D, y_D, t) = \sum_{(x,y) \in N(x_D, y_D)} I(x, y, t) \tag{5}$$

Both GAs and DAs are modeled by fuzzy granules [18]. In particular, an s-shape membership function is adopted for the GA:

$$\mu_{GA}(I_{GA}(x_G, y_G, t)) = \begin{cases} 0 & \text{if } I_{GA}(x_G, y_G, t) \le a \\ 2\left(\dfrac{I_{GA}(x_G, y_G, t) - a}{b - a}\right)^2 & \text{if } a \le I_{GA}(x_G, y_G, t) \le \dfrac{(a+b)}{2} \\ 1 - 2\left(\dfrac{b - I_{GA}(x_G, y_G, t)}{b - a}\right)^2 & \text{if } \dfrac{(a+b)}{2} \le I_{GA}(x_G, y_G, t) \le b \\ 1 & \text{if } I_{GA}(x_G, y_G, t) \ge b \end{cases} \tag{6}$$

where parameters a and b control the curve slope of the s-function. The parameters a and b have to be chosen appropriately so as to make the granulation process independent of the number of users involved in the collaborative situation. As explained in [16], we set $a = I_{GA}^{min}(x_G, y_G, t)$ and $b = \frac{2}{3} \cdot I_{GA}^{max}(x_G, y_G, t)$.

The choice of b is motivated by the following reasonable assumption: a grouping event occurs when at least half of the U users are close to each other. Thus, we consider that when a number of users higher than 2/3 U are close to each other, then the grouping event should have maximum degree. The minimum value $a = I_{GA}^{min}(x_G, y_G, t)$ corresponds to the case in which a unique user has left a mark in the squares $N(x_G, y_G)$. If we assume that $\delta = 0.5$ (the value used in our experiments), then the minimum value is $I_{GA}^{min}(x_G, y_G, t) = 5 \cdot I_{MAX}$. The maximum value $I_{GA}^{max}(x_G, y_G, t)$ corresponds to the case in which all the users are still and leave marks on the same squares. As proved in [16], $I_{GA}^{max}(x_G, y_G, t) = 5 \cdot U \cdot I_{MAX} \cdot 1/(1-\alpha)$, where U is the number of users. Fig. 1(a) shows an example of a GA fuzzy granule.

Since the value of b depends on the number of users, the result of the granulation process is independent of U. As regards DA, the following z-shaped membership function is adopted as fuzzy granule:

$$\mu_{DA}(I_{DA}(x_D, y_D, t)) = \begin{cases} 1 & \text{if } I_{DA}(x_D, y_D, t) \le a \\ 1 - 2\left(\dfrac{I_{DA}(x_D, y_D, t) - a}{b - a}\right)^2 & \text{if } a \le I_{DA}(x_D, y_D, t) \le \dfrac{(a+b)}{2} \\ 2\left(\dfrac{b - I_{DA}(x_D, y_D, t)}{b - a}\right)^2 & \text{if } \dfrac{(a+b)}{2} \le I_{DA}(x_D, y_D, t) \le b \\ 0 & \text{if } I_{DA}(x_D, y_D, t) \ge b \end{cases} \tag{7}$$

where parameters a and b have the same values computed for $I_{GA}(x_G, y_G, t)$. Unlike GA, which considers a group of users, DA takes only one user into consideration: a disjoining event occurs when a user is alone in the area of the mark. Thus, $a = 5 \cdot I_{MAX}$ and $b = 2 \cdot 5 \cdot I_{MAX} \cdot /(1-\alpha)$, where b coincides with the maximum value achievable in correspondence to two users still and alone. Obviously, for values higher than b, we can be sure that the user is not alone and therefore the disjoining event is recognized with minimum certainty degree. Fig. 1(b) shows an example of a DA fuzzy granule. For simplicity of notation, in the following we denote the certainty degrees $\mu_{GA}(I_{GA}(x_G, y_G, t))$ and $\mu_{DA}(I_{DA}(x_D, y_D, t))$ as $GR(t)$ and $DJ(t)$, respectively.

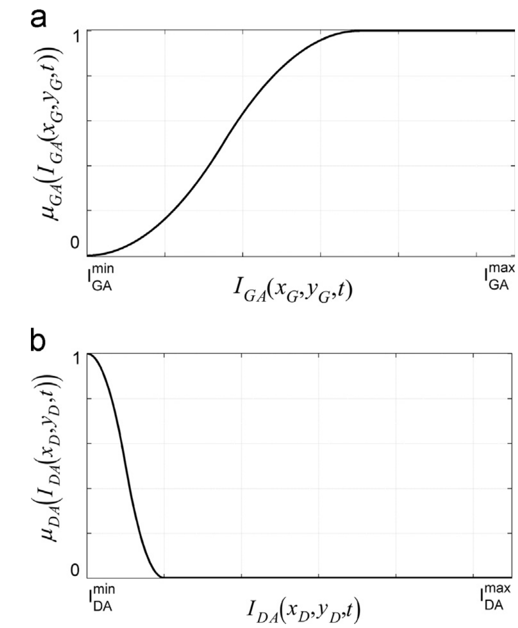


Fig. 1. Membership functions used to model the granulation process of the GA (a) and the DA (b).

The control logic of a *GA* can be summarized as follows:

Loop
Wait for T_M seconds;
Get the positions (x_P, y_P) of all the *MAs* of the group;
Compute position (x_G, y_G) as center of gravity of the positions (x_P, y_P);
Get from the Environment the intensity $I(x, y)$ of each square in the neighboring area $N(x_G, y_G)$;
Compute the intensity $I_{GA}(x_G, y_G)$ by formula (4);
Compute the certainty degree of the grouping event by formula (6);
End loop

The control logic of a *DA* can be summarized as follows:

Loop
Wait for T_M seconds;
Get from the Environment the intensity $I(x, y)$ of each square in the neighboring area $N(x_D, y_D)$;
Compute the intensity $I_{DA}(x_D, y_D)$ by formula (5);
Compute the certainty degree of the grouping event by formula (7);
End loop

In conclusion, the processing performed by the first two levels can be summarized as follows. Information granules originate at the marking processing level where the environment, a short-term memory of the user′s positions, supports an emergent process activated by the users′ proximity. Indeed, information provided by the mark intensity allows identifying groups of users who have been close to each other in a recent period of time. Subsequently, in order to extract essential information from mark intensity, the fuzzy granulation processing emphasizes the intensity values that are more relevant for identifying the grouping and disjoining events. The membership functions, which characterize the fuzzy granules, are automatically calibrated on the number of participants to the event and therefore their semantics results in practice to be independent of this number.

2.3. The fuzzy inference processing level

This level is in charge of assessing the current user's situations. It is accomplished by a Situation Agent (SA) which is aimed at recognizing four types of situations related to collaboration: (i) pre-collaboration (*PreC*), while the user is discussing with one or more other users about the coming collaboration; (ii) on-going collaboration (*OngC*), while the user is attending the collaboration; (iii) collaboration pause (*PauC*), while the user is having a break during the collaboration; (iv) post-collaboration (*PstC*), while the user is discussing with one or more other users about the collaboration, once it has terminated.

The SA uses fuzzy rules to provide for each user the certainty degree of being in each situation. The methodology used for identifying the fuzzy rules is expert-driven, and can be summarized as follows. In the first step, we identify the situations of interest for the specific application domain by interviewing the end-users. Through the identification of the situations, we define the output variables of the system and their possible values. In the second step, we characterize the relations between the system outputs and some contextual information, which in the case in point is the degree of interaction among participants. In the third step, we design a general-purpose ontology as suitable contextual source of information for deriving the above situations. With respect to the application domain, the basic assumption is that grouping and disjoining events are necessary, but not sufficient because they are based on proximity information only. In the fourth step, we identify the antecedent part of the rules, by expressing some precise form of knowledge on the system modeling. As regards this step, contextual information of proximity has been combined with information about the situations defined in the previous steps, in order to distinguish pre-collaborations and post-collaborations situations.

Specifically, fuzzy rules have been designed so as to describe the constraints characterizing the sequence of situations occurring during a collaboration. In this way, a small set of fuzzy rules was defined to recognize each situation for the *i*-th user, for a total number of 14 rules listed in Table 1. The consequent part of each rule expresses the certainty degree of being in one situation (Low or High). The antecedent part of each rule is a conjunction of fuzzy terms which express contextual information provided by the GA and the DA as well as the information about the situation recognized at the previous step. This enables the SA to accomplish the sequence $PreC \rightarrow OngC \rightarrow (PauC \rightarrow OngC) \rightarrow PstC$ for the *i*-th user. A small memory, called Agenda, stores at each instant the certainty degree of the situation recognized for the *i*-th user. At each instant $t = \bar{t} + T_D, \bar{t} + 2T_D, \ldots$, the SA detects a new situation for the *i*-th user by taking into account both the content of the Agenda at step $t - T_D$ and the certainty degree $GR(t)$ of the grouping event provided at time t by the GA to which the *i*-th user belongs. The variable $GR(t)$ is described by three linguistic values (Low, Medium and High) defined by Gaussian fuzzy sets. Fuzzy rules aiming to recognize the *PauC* situation use also the certainty degree *DJ* of the disjoining event provided at time t by the DA corresponding to the *i*-th user. The variable $DJ(t)$ is described by two linguistic values (Low and High) defined by Gaussian fuzzy sets.

Table 1
Fuzzy rules used by the SA for situation recognition.

Rules for *PreC* recognition
R_1 IF GR(t) IS M AND PreC($t-T_D$) IS L THEN PreC(t) IS H
R_2 IF GR(t) IS H AND PreC($t-T_D$) IS H THEN PreC(t) IS L
Rules for *OngC* recognition
R_3 IF GR(t) IS H AND PreC($t-T_D$) IS H THEN OngC(t) IS L
R_4 IF GR(t) IS H AND OngC($t-T_D$) IS L THEN OngC(t) IS H
R_5 IF GR(t) IS H AND OngC($t-T_D$) IS H THEN OngC(t) IS H
R_6 IF GR(t) IS M AND OngC($t-T_D$) IS H THEN OngC(t) IS L
R_7 IF GR(t) IS H AND DJ(t) IS L AND PauC($t-T_D$) IS H THEN OngC(t) IS H
Rules for *PstC* recognition
R_8 IF GR(t) IS M AND OngC($t-T_D$) IS H THEN PstC(t) IS L
R_9 IF GR(t) IS M AND PstC($t-T_D$) IS L THEN PstC(t) IS H
R_{10} IF GR(t) IS L AND PstC($t-T_D$) IS H THEN PstC(t) IS L
R_{11} IF GR(t) IS L AND PauC($t-T_D$) IS L THEN PstC(t) IS L
Rules for *PauC* recognition
R_{12} IF GR(t) IS H AND DJ(t) IS H AND OngC($t-T_D$) IS H THEN PauC(t) IS L
R_{13} IF GR(t) IS H AND DJ(t) IS H AND PauC($t-T_D$)IS L THEN PauC(t) IS H
R_{14} IF GR(t) IS H AND DJ(t) IS L AND PauC($t-T_D$) IS H THEN PauC(t) IS L

The control logic of an *SA* can be summarized as follows

Loop
  Wait for T_D seconds;
For each user i
    Get from the *GA* of the user i the current grouping degree *GR*;
    Get from the *DA* of the user i the current disjoining degree *DJ*;
    Get from the Agenda of the user i the degrees of the situations recognized at the previous step;
    Infer from rules in Table 1 the degrees of the current situations of user i;
    Update the Agenda with the degrees of the current situations of user i;
  End for
End loop

The parameters of fuzzy sets used in the SA rules were defined in a completely automatic way by means of a neuro-fuzzy learning process. In particular, fuzzy rules were mapped to a neural architecture, resulting in a neuro-fuzzy network comprising five layers of neurons (Fig. 2). Neurons in layer 1 simply provide input values to the network. Neurons in layer 2 represent fuzzy sets defined on the input variables: each neuron receives the input value and computes a membership value through a Gaussian function. Each neuron in layer 2 has two adjustable parameters corresponding to the center and the width of the Gaussian function. Each neuron in layer 3 corresponds to a rule: it is connected to the neurons in layer 2 which implement the fuzzy sets used in the antecedent of the rule and computes the activation strength as product of the membership values output by these neurons. The neurons in layer 3 have no adjustable parameter. Neurons in layer 4 represent fuzzy singletons used in the consequent part of fuzzy rules. Consequent fuzzy singletons are adjustable parameters of the network. Layer 5 performs defuzzification and includes one neuron for each considered situation. The outputs of these neurons are certainty degrees of being in the corresponding situation.

Parameters of neurons in layers 2 and 4 are automatically defined via supervised learning using the same algorithm defined

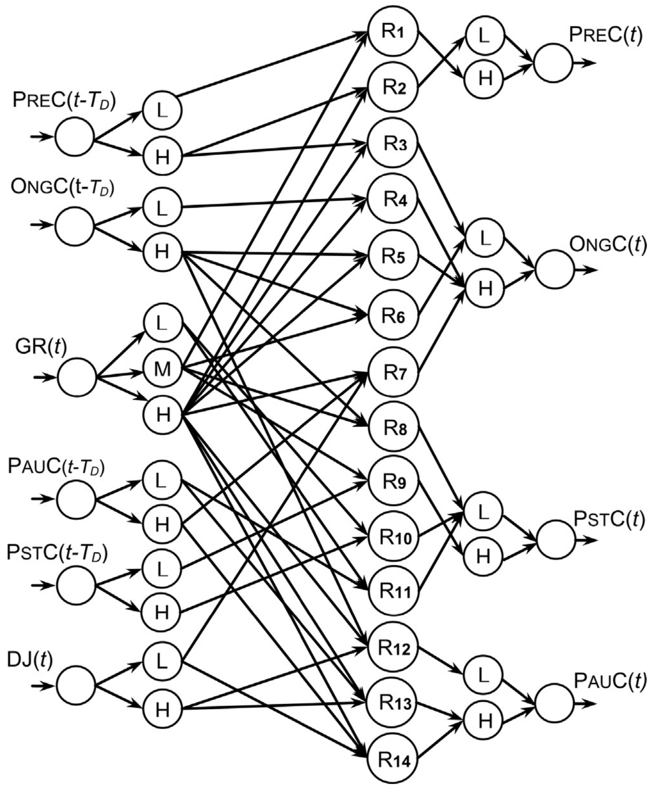


Fig. 2. Architecture of the adopted neuro-fuzzy network.

for the well-known ANFIS model [19]. The ANFIS learning algorithm is a two-step hybrid procedure that combines the back-propagation gradient descent method and the least squares method. In the first step, consequent parameters are updated via a least squares method. In the second step the error rates propagate backward into the layers and the premise parameters are updated by the gradient descent.

By fuzzy rule inference, the SA can provide a certainty degree for each situation and for each user at each time step. For example, in the preliminary phase of a meeting, a user may be in the *PreC* situation with high degree and in the *OngC* situation with low degree; likewise, during the ending phase of a meeting a user may be in the *OngC* situation with low degree and in the *PstC* situation with high degree. Given the certainty degrees of all situations for each user at a certain time step, the SA selects the situation with the highest degree as current situation to be included in the Agenda of the user.

3. Architecture and knowledge representation in the CSA system

A robust and general approach to CSA should guarantee that system architecture and behavioral knowledge can be easily integrated in an open environment. Further, a variety of contextual, possibly uncertain, collective inputs should be supported. Finally, situational knowledge should be provided to multiple applications. To this aim, the architecture of the system has been designed in compliance with an agent-oriented approach [20–22], which operates at the knowledge level, shows flexible behavior, easy maintenance, reusability and platform independence.

This is achieved thanks to the use of highly standardized technologies, such as Semantic Web and Approximate Reasoning [14,23]. In the following, we will describe the main modules of the architecture and their interaction and some aspects of the knowledge representation.

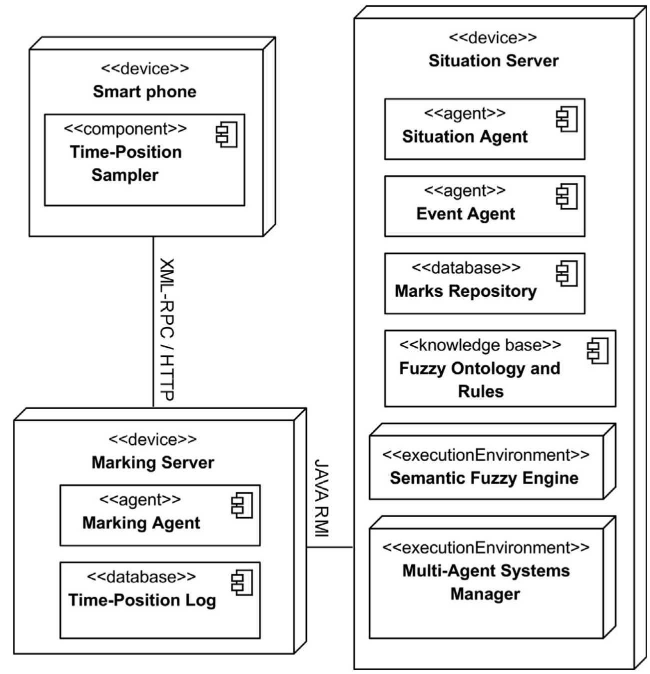


Fig. 3. Overall system architecture.

3.1. Main architectural modules and their interaction

Fig. 3 shows a UML deployment diagram of the proposed system. Here, there are three device categories, i.e., *Smart phone* on the client side, *Marking Server* and *Situation Server* on the server side. The Smart phone provides the server side with the current position of the user, generated by its *Time-Position Sampler* module. Position estimation can be based on a GPS signal reader, or can be computed by means of other technologies, such as GSM and Wi-Fi [24].

On the server side, the Marking Server manages the marking process, i.e., it hosts the MA and the *Time-Position Log* module, and delivers marks to the *Situation Server*. Finally, the Situation Server manages the Environment (via a *Multi-Agent Systems Manager*), hosts the SA and EA instances, and supports both fuzzy granulation and situation fuzzy inference processes, according to linguistic variables and rules processed by means of the *Fuzzy Engine*. A single Marking Server can support many smart phone clients, via a lightweight and platform-independent communication protocol based on XML-RPC over HTTP. Thus, any client-side platform can be easily integrated with the system. A single Situation Server can support many Marking servers, via an efficient Java-RMI communication protocol. Indeed server-side subsystems are entirely Java-based. More specifically, the following environments have been employed to develop and execute the infrastructure. The Semantic Web Engine is based on Apache Jena,[1] a Java framework for building Semantic Web applications, used in conjunction with Pellet,[2] a Java based OWL DL reasoner. The Fuzzy Engine is based on jFuzzyLogic,[3] a Java package that implements a series of basic fuzzy operations as well as a fuzzy inference system. Finally, the Multi-Agent Systems Manager is based on Repast Simphony,[4] a Java-based modeling system supporting the development of interacting agents. It can be used as a GUI-based (user driven) simulation environment, as well as an execution engine run from another Java application.

Fig. 4 shows a scenario of communication among the most important modules, by using the UML communication diagram.

[1] http://incubator.apache.org/jena.
[2] http://clarkparsia.com/pellet.
[3] http://jfuzzylogic.sourceforge.net.
[4] http://repast.sourceforge.net.

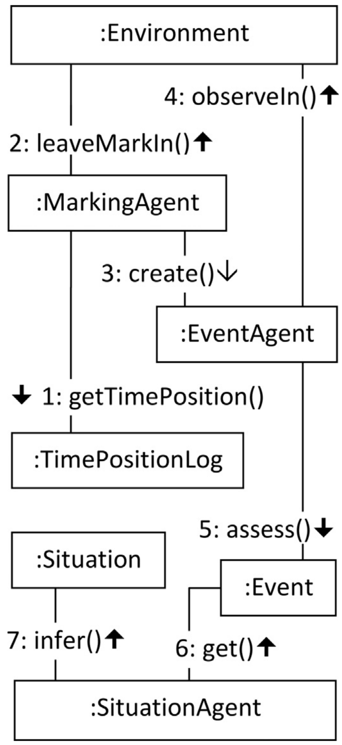


Fig. 4. Communication diagram for the situation reasoning process.

The interaction starts with the MA (1), which gets the position from the *TimePositionLog* module and leaves a mark in the *Environment* (2). Supposing that the MA receives from the *Environment* the information whether the mark is going to be superimposed on marks left by other MAs, the MA creates an EA (3). Possibly, EAs corresponding to the same group of users are fused. An EA performs a local observation in the *Environment* (4), and assesses a specific type of event (5). Finally, an SA takes as input the specific event (6) and infers the situation (7).

3.2. Knowledge representation in the CSA system

CSA relies on a distributed system. Hence, knowledge portability, integration and extensibility are key features since context reasoning implies collaboration among software agents that manage their own contextual sources. For this purpose, our system employs web knowledge representation standards, such as Semantic Web Languages [25] and Fuzzy Markup Language [26].

In the Semantic Web domain, the Resource Description Framework (RDF)[5] and the Web Ontology Language (OWL)[6] are the basic languages traditionally employed to author ontologies [27]. RDF and OWL are W3C standard specifications, well-supported by semantic engines. In order to manage fuzzy information in an OWL compliant ontology, we used a representation pattern proposed in [14,28]. The pattern, named Fuzzy Ontology Representation (FOR), considers a fuzzy property as a relation between two concepts, representing additional attributes to describe each relation instance. It is applicable to properties that are related to the same base variable and to the same pair of concepts. More specifically, in the FOR pattern an OWL group of properties is transformed into a concept, which includes a specification of the degree for each property. In other words, we assert that there

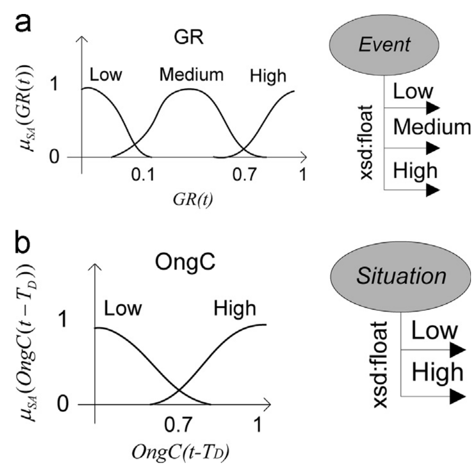


Fig. 5. Concrete representation of the fuzzy property *GR*(*t*) (a) and *OngC*(*t*) (b) with the FOR pattern.

is a property with a certain degree. Each degree is the membership level of the base variable to a specific fuzzy set. It is worth noting that this scheme can be used also in case of a property related to a single concept. In such case, the concept property corresponds to the concept itself. As an example, Fig. 5(a) and (b) shows the representation of the fuzzy properties concerning the certainty degrees *GR*(*t*) and *OngC*(*t*) of the grouping event and the certainty degrees of the *OngC* Situation. Since FOR pattern is RDF and OWL compliant, it is possible to extend any property with fuzzy characters using conventional RDF/OWL engines.

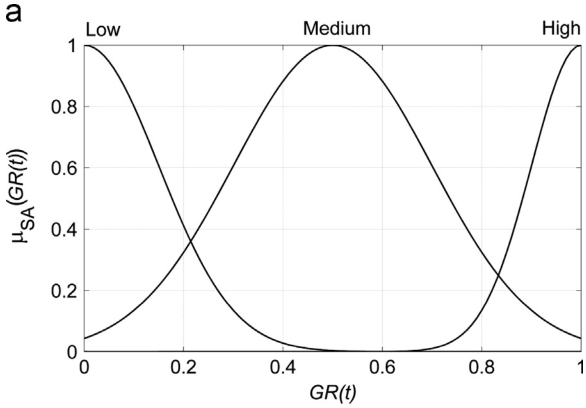
Context Awareness should enable users to seamlessly employ and configure the intelligent devices and systems in their environments without being cognitively and physically overloaded. The complexity associated with the number, varieties, and uses of smart personal devices, and of possibly different services, requires a technology that lets intelligence embody in the environment without interfering with the user's task. In other words, Context Awareness requires necessarily two types of intelligence, i.e., reasoning and interoperability. Our proposal exploits fuzzy-based reasoning and position-based stigmergy to realize collaborative situation aware services.

The Fuzzy Markup Language (FML) is used to enable the interoperable fuzzy-based reasoning, since it describes both the data base and rule base [26,29]. FML is an XML-based language used to model fuzzy controllers. It provides a platform-independent grammar over shared resources. FML is particularly suitable for: (i) distributing the fuzzy control flow, in order to minimize the global deduction time and to better exploit the natural distributed knowledge repositories; (ii) acquiring, online, the user's behavior and environment status, in order to apply context-aware adaptivity [26]. A simple example of FML fuzzy knowledge base is shown in Fig. 6. Here, Fig. 6(a) shows the definition of the membership functions associated with the linguistic terms "Low", "Medium", and "High" of the linguistic variable *GR*(*t*), expressed in the usual graphical representation. Fig. 6(b) is the corresponding FML serialization.

The user's position is detected by the Time-Position Sampler at the client side. Whatever the positioning mechanism employed, the server is expected to receive the position of the user. In the CSA system, the GPX (GPS eXchange format)[7] is used to express interoperable tracks as ordered collections of points where the user has been, and corresponding timestamps. GPX is an open and widely used XML format that allows describing waypoints, tracks

[5] http://www.w3.org/RDF.
[6] http://www.w3.org/OWL.
[7] http://www.topografix.com/gpx.asp.

a



b

```
<!DOCTYPE FUZZYCONTROL SYSTEM "fml.dtd">
<FUZZYCONTROL ip = "localhost">
 <KNOWLEDGEBASE>
  <FUZZYVARIABLE  domainleft = "0"
                  domainright = "1"
                  name = "GR"
                  scale = "dimensionless"
                  type = "INPUT">
   <FUZZYTERM name = "Low">
    <GAUSSIANSHAPE param1 = "0"
                   param2 = "0.15">
    </GAUSSIANSHAPE>
   </FUZZYTERM>
   <FUZZYTERM name = "Medium">
    <GAUSSIANSHAPE param1 = "0.5"
                   param2 = "0.2">
    </GAUSSIANSHAPE>
   </FUZZYTERM>
   <FUZZYTERM name = "High">
    <GAUSSIANSHAPE param1 = "1"
                   param2 = "0.1">
    </GAUSSIANSHAPE>
   </FUZZYTERM>
  </FUZZYVARIABLE>
 </KNOWLEDGEBASE>
<FUZZYCONTROL>
```

Fig. 6. Classical visual definition of the fuzzy linguistic variable $GR(t)$ (a), and its FML serialization (b).

```
<?xml version="1.0" encoding="UTF-8">
<gpx xmlns="http://www.topografix.com/GPX/1/1">
  <metadata>
    <name>Scenario C</name>
    <desc>Meeting at the Department</desc>
    <author>CSA System</author>
    <time>2012-10-19T9:15:41Z</time>
  </metadata>
  <trk>
    <name>User P3</name>
    <trkseg>
      <trkpt lat="43.35382" lon="10.45587">
       <time>2012-10-19T9:16:5Z</time>
      </trkpt>
      <trkpt lat="..." lon="...">
       <time>...</time>
      </trkpt>
         ...
    </trkseg>
  </trk>
</gpx>
```

Fig. 7. A fragment of a GPX waypoint containing the user position.

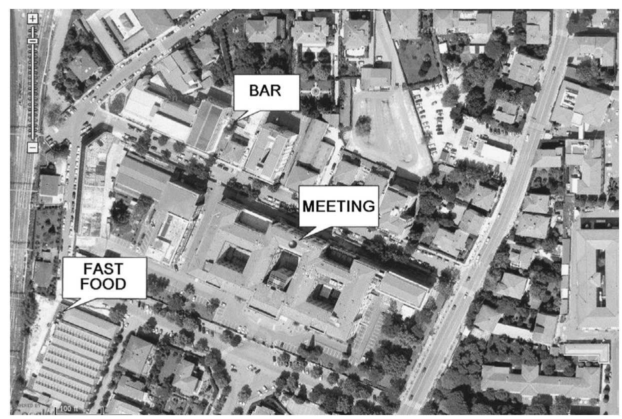


Fig. 8. Points of main interest.

and routes. An example of a GPX track sent to the server is shown in Fig. 7. Here, the most important elements are: metadata with name, description, author and timestamp of the track, as well as the track itself, with a name, a series of geographic coordinates and their timestamps.

4. Simulation results

To assess the effectiveness of the proposed multi-agent system in detecting collaboration situations, we tested our scheme on four real-world scenarios involving a different number U of participants ($P_1, \ldots, P_U$). The four scenarios, denoted as A, B, C and D, refer to a meeting among 10, 8, 6 and 4 participants, respectively. Scenarios are characterized as follows:

- Scenario A ($U=10$). P_1 meets P_2 at a bar before arriving at the meeting place. P_8 reaches P_1 and P_2 at the bar and then together they go to the meeting place. During the meeting, P_3 leaves the meeting place for a short time to go to the bar. Further, P_4 and P_5 leave the meeting place for a longer time to go to the fast food. P_1, P_2, P_3 and P_4 leave the meeting place before the other participants.
- Scenario B ($U=8$) was obtained by selecting participants $P_1, \ldots, P_8$ from the scenario A.
- Scenario C ($U=6$) was obtained by selecting participants $P_1,\ldots, P_6$ from the scenario A.
- Scenario D ($U=4$) was obtained by selecting participants $P_1,\ldots, P_4$ from the scenario A.

Fig. 8 shows the points of main interest in the considered scenarios. As an example, in Fig. 9 we show the GPS data generated by the mobile devices of each participant in scenario B.

As a first step, continuous GPS data have been discretized into a grid of square cells. We adopted a 100×100 grid. More specifically, for all the scenarios the system parameters were set as follows: $L=100$, $\delta=50\%$, $\alpha=0.5$, $T_D=T_M=T=60$ s. We considered a time interval of 100 T_D.

As an example, in Fig. 10 we show the value of marking intensities corresponding to different steps of the scenario C. Here, x and y are dimensionless integer coordinates used to refer to cells of the physical area under observation, which has been discretized into a grid. More specifically, in Fig. 10(a) at time step $t=27$ all participants are moving alone and far from the meeting place. In particular, the two participants located approximately at ($x=60$, $y=60$) are moving close to each other. For this reason their marks are slightly overlapping, although they are moving fast and therefore their marks are quite small. Taller marks correspond to participants that are moving slower, e.g., the participants located approximately at (30, 40) and

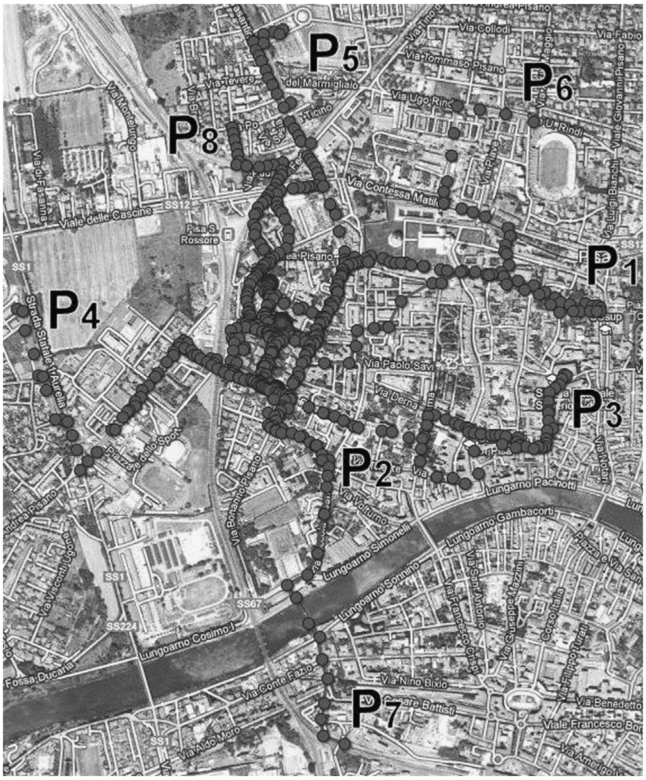


Fig. 9. GPS data for scenario B involving 8 participants to a meeting.

(20, 50). Very small marks correspond to fast moving participants, e.g., the remaining participants. In particular, for the participant located at (20, 80) we can identify also the direction of the movement. Fig. 10(b) represents the scenario at instant $t=45$, when two late participants are approaching the meeting place, thus creating a minor accumulation of marks. Here, the tallest mark intensity corresponds to the meeting place, where the other participants are already located. In Fig. 10(c), corresponding to instant $t=72$, all participants are still at the meeting place, thus creating a unique and tall mark intensity. Finally, Fig. 10(d) is taken at $t=78$ when some participants leave the meeting place for a break. Here the minor intensity peak corresponds to the break location, whereas the tallest mark corresponds to the meeting place.

Next, marking intensities have been employed as described in Section 2.2 to calculate the certainty degrees $GR(t)$ and $DJ(t)$ for the grouping and the disjoining events, respectively. As an example, in Fig. 11(a) we show the certainty degree $GR(t)$ computed for the grouping event in correspondence of $t=49$, when all participants reach the meeting place. It can be seen that in this step, for all participants, $GR(t)$ is maximum. Fig. 11(b) shows $GR(t)$ at time step $t=55$. We observe that there exist two peaks: the highest is in correspondence to the meeting place while the lowest is in correspondence to the positions of participants P_4 and P_5, who are leaving the group.

In order to train the neuro-fuzzy system we have adopted the samples of the scenario A. To assess the goodness and stability of the neuro-fuzzy approach we have employed a 10-fold cross-validation. The learning process is stopped after 100 epochs or when the network error drops below a small fixed threshold. In our trials,

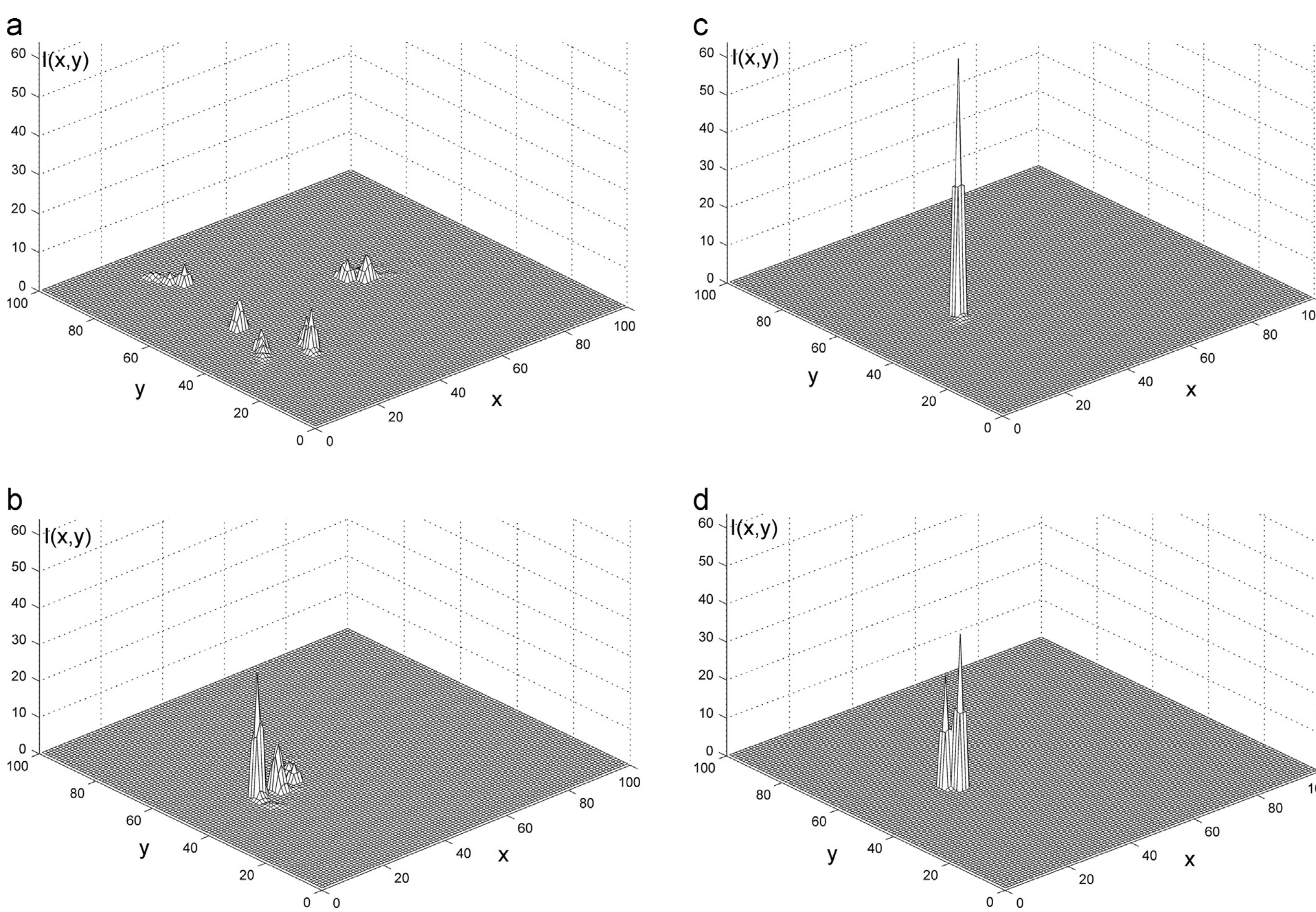


Fig. 10. Marking intensities in different steps: initially (a), during *PreC* (b), *OngC* (c) and *PauC* (d) situations.

the minimum training error was set to 0 and the initial learning rate to 0.01. Moreover, during the training, the learning rate was updated by considering decreasing/increasing multiplicative factors equal to 0.9 and 1.1 respectively. We observed very stable results in terms of the obtained classification error (misclassified situations at each considered time instant t) that was equal to 2.5% in all runs. We

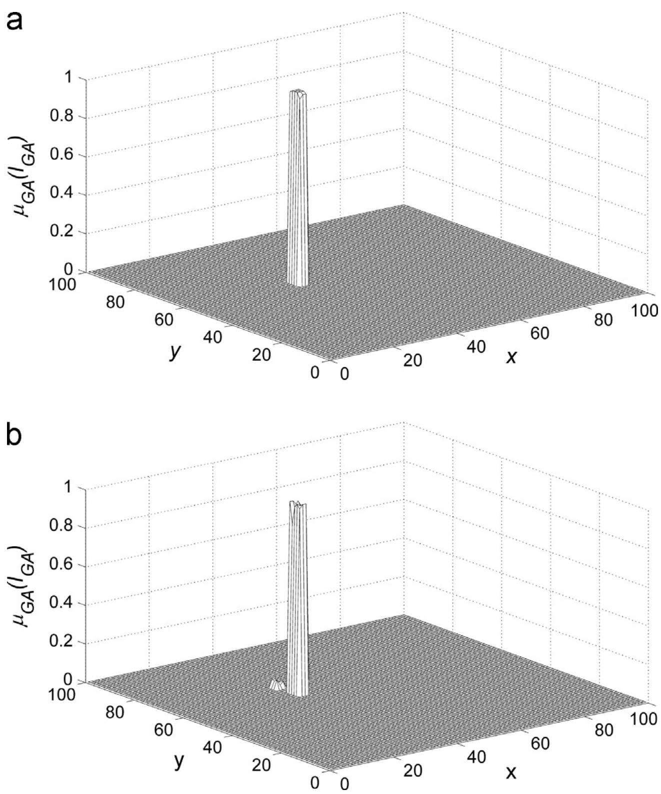


Fig. 11. Certainty degrees $GR(t)$ when all the participants are still at the same location (a) and when two participants are left the group for a pause (b).

chose randomly a knowledge base generated in one of the trials and used it in SA for recognizing situations in the other scenarios. In Fig. 12, we show the partitions of each linguistic variable after the neuro-fuzzy learning.

To prove the effectiveness of our approach, for each scenario we compared the instants when the SA recognized the beginning and the end of each situation to the instants when the beginning and the end of the situation actually occur. Tables 2, 3, 4 and 5 compare these instants for scenarios A, B, C and D, respectively. The values reported in each table represent the instants corresponding to the beginning and the end of each situation. The values in brackets represent the instants in which each situation actually begin/ends. The values out of brackets represent the instants in which the SA recognizes the beginning/end of each situation. The closer these values are, more precisely the SA recognizes the instants in which each situation begins/ends.

As it can be observed in Table 2, the SA detected almost exactly the instants when each situation begins and ends. Only for users

Table 2
Comparisons between instants recognized by the SA and actual instants (between parentheses) for the beginning and the end of each situation in scenario A.

Participant	*PreC* begins	*PreC* ends/*OngC* begins	*PauC* begins	*PauC* ends	*OngC* ends /*PstC* begins	*PstC* ends
P_1	28(28)	46(46)	78(78)		82(82)	90(90)
P_2	28(28)	46(46)	78(78)		82(82)	90(90)
P_3		48(48)	61(61) 80(80)	67(67)	82(82)	90(90)
P_4	41(41)	46(46)	53(53) 77(78)	71(71)	82(82)	90(90)
P_5	44(44)	46(46)	54(54) 81(82)	71(71)	82(82)	90(90)
P_6	44(44)	46(46)	81(82)		83(83)	90(90)
P_7	38(38)	46(46)	81(82)		83(83)	90(90)
P_8	29(29)	47(47)			82(82)	90(90)
P_9	45(45)	47(47)			82(82)	90(90)
P_{10}	46(46)	47(48)			82(82)	90(90)

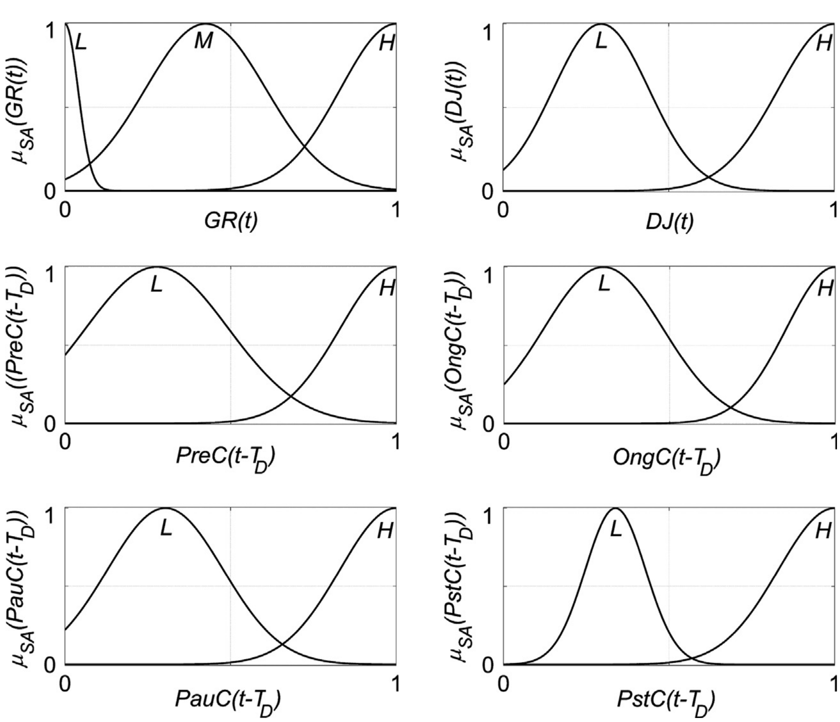


Fig. 12. Partitions of the linguistic variables used in the rules after the neuro-fuzzy learning.

Table 3
Comparisons between instants recognized by the SA and actual instants (between parentheses) for the beginning and the end of each situation in scenario B.

Participant	*PreC* begins	*PreC* ends/ *OngC* begins	*PauC* begins	*PauC* ends	*OngC* ends/ *PstC* begins	*PstC* ends
P_1	28(28)	45(46)	78(78)		82(82)	90(90)
P_2	28(28)	46(46)	78(78)		82(82)	90(90)
P_3		48(48)	61(61) 80(80)	67(67)	82(82)	90(90)
P_4	41(41)	45(46)	53(53) 77(78)	71(71)	82(82)	90(90)
P_5	44(44)	46(46)	54(54) 81(81)	71(71)	82(82)	90(90)
P_6	44(44)	45(46)	82(82)		83(83)	90(90)
P_7	38(38)	45(46)	81(81)		83(83)	90(90)
P_8	29(29)	46(47)	82(82)		83(83)	90(90)

Table 4
Comparisons between instants recognized by the SA and actual instants (between parentheses) for the beginning and the end of each situation in scenario C.

Participant	*PreC* begins	*PreC* ends/ *OngC* begins	*PauC* begins	*PauC* ends	*OngC* ends/ *PstC* begins	*PstC* ends
P_1	28(28)	45(46)	78(78)		83(83)	90(90)
P_2	28(28)	45(46)	78(78)		83(83)	90(90)
P_3		48(48)	61(61) 80(80)	67(67)	83(83)	90(90)
P_4	41(41)	45(46)	53(53) 78(78)	69(70)	83(83)	90(90)
P_5	44(44)	46(46)	54(54) 81(81)	69(70)	83(83)	90(90)
P_6	44(44)	45(46)	82(82)		83(83)	90(90)

Table 5
Comparisons between instants recognized by the SA and actual instants (between parentheses) for the beginning and the end of each situation in scenario D.

Participant	*PreC* begins	*PreC* ends/ *OngC* begins	*PauC* begins	*PauC* ends	*OngC* ends/ *PstC* begins	*PstC* ends
P_1	28(28)	29(28)	78(80)		81(81)	82(82)
P_2	28(28)	29(28)	78(78)		81(81)	82(82)
P_3		48(48)	61(61)	66(67)	81(81)	82(82)
P_4	41(41)	42(42)	53(53) 78(78)	70(70)	81(81)	82(82)

Table 6
Responsiveness values for the four scenarios and the four situations.

Scenario	*PreC* begins	*PreC* ends/ *OngC* begins	*PauC* begins	*PauC* ends	*OngC* ends/ *PstC* begins	*PstC* ends	**avg**
A	0.0	0.1	0.2	0.0	0.0	0.0	**0.05**
B	0.0	0.625	0.0	0.0	0.0	0.0	**0.11**
C	0.0	0.67	0.0	0.33	0.0	0.0	**0.17**
D	0.0	0.5	0.5	0.0	0.0	0.0	**0.21**

Table 7
Classification rates obtained in the four scenarios.

Tuning method	**A** (%)	**B** (%)	**C** (%)	**D** (%)
SA with fuzzy rule parameters tuned by neuro-fuzzy learning	97.5	97.0	97.5	96.0
SA with fuzzy rule parameters tuned heuristically [16]	97.5	95.0	93.5	91.0

Table 8
Average responsiveness values for the four scenarios obtained by tuning the fuzzy rule parameters through the neuro-fuzzy learning and heuristically.

Tuning method	**A**	**B**	**C**	**D**
SA with fuzzy rule parameters tuned by neuro-fuzzy learning	0.05	0.11	0.17	0.21
SA with fuzzy rule parameters tuned heuristically [16]	0.06	0.20	0.27	0.33

P_4, P_5, P_6 and P_7 the SA recognized the beginning of the *PstC* situation with a step in advance with respect to the target step. Moreover, the SA recognized with a step in advance the time when the user P_{10} reached the other users in the meeting place (i.e., beginning of *OngC*). For the scenario B (Table 3) the SA detected the beginning of *OngC* with a step in advance for some users. The beginning and the end of all the other situations are correctly detected. Similar observations can be made by analyzing the results listed in Tables 4 and 5.

The effectiveness of the SA agent in recognizing situations was measured through the responsiveness index defined as follows:

$$R(S_s) = \frac{\sum_{i=1}^{U} |t_{i,s} - t'_{i,s}|}{U} \tag{8}$$

where $t_{i,s}$ represents the time step at which the s-th situation begins/ends for each i-th participant and $t'_{i,s}$ is the time step at which the SA agent recognizes the beginning/end of the situation. The lower the responsiveness value is, the higher the accuracy of the system is to recognize the beginning/end of the situation. In Table 6 we show the responsiveness values obtained in correspondence of the beginning/end of each situation recognized during the test of the system in the four scenarios. It can be seen that on an average the value of the responsiveness is close to 0. Further, for all the situations, the responsiveness is lower than one time step (except for the end of the *PstC* situation, where responsiveness value is 1). This demonstrates the good performance of the SA in detecting situations.

A further consideration concerns the ability of the SA to recognize situations not only in correspondence of the beginning/end steps but also in the middle steps when situations continue. We observed that the SA succeeds in detecting the correct situation in all time steps except for a very low number of cases, as demonstrated by the classification rates obtained in the four considered scenarios (Table 7).

Finally, the proposed system was compared with the system presented in [16] where fuzzy rule parameters were tuned heuristically by hand. Table 8 shows the average responsiveness values obtained by the two systems in correspondence of the four considered scenarios. As it can be observed, the system where fuzzy rule parameters were tuned by the neuro-fuzzy learning identifies more precisely the beginning and the end of each situation by obtaining for all the considered scenarios average responsiveness values lower than the system where fuzzy rule parameters were heuristically tuned.

Further, we compared the two systems also in terms of the obtained classification rates. For each considered scenario, the classification rate was computed by adopting a classical metric, namely, the percentage of correctly classified samples (i.e., the percentage of samples for which the output of the system matches with the available target.

Table 9
Table of abbreviations.

Abbreviated name	Full name	Description
CSA	Collaborative Situation Awareness	A computing paradigm that enables applications to sense and explore collaborative situations in which the users are
DA	Disjoining Agent	An event agent responsible for detecting disjoining events, which occur when a user leaves the group
DJ	Disjoining degree	Certainty degree of the disjoining event
EA	Event Agent	An agent which detects events by observing marks in the environment
GA	Grouping Agent	An event agent responsible for detecting grouping events, which occur when users are close to each other
GR	Grouping degree	Certainty degree of the grouping event
I	(Mark) Intensity	Mark Intensity associated with a given position in the Environment at a given instant
MA	Marking Agent	An agent which periodically leaves a mark in the environment at the position where the user is currently located
N	Neighborhood	Neighboring area in the Environment, centered in a given position
OngC	On-going collaboration	A situation occurring when the user is attending the collaboration
PauC	Collaboration pause	A situation occurring when the user is having a break during the collaboration
PreC	Pre-collaboration	A situation occurring when the user is discussing with one or more users about the coming collaboration
PstC	Post-collaboration	A situation occurring when the user is discussing with one or more other users about the collaboration, which has just terminated
Q	Square	A square of the grid superimposed on the spatial area under observation
SA	Situation Agent	An agent responsible for assessing the current user's situation

In Table 7 we show classification rates obtained by the two systems for all the four scenarios A, B, C and D. We may note that the neuro-fuzzy approach outperforms the approach based on heuristic rules. In effect, the latter achieves classification rates lower for the four scenarios than the former.

In conclusion, the new proposed approach allows tuning the fuzzy sets effectively and automatically, thus avoiding the burden of manual tuning. For better readability Table 9 summarizes all the abbreviations used throughout the paper.

5. Conclusions and future work

In this paper, we presented a multi-agent system for the detection of situations related to social events via position-based stigmergy and neuro-fuzzy learning. The proposed system is structured into three different processing levels managed by different agents in order to recognize situations through inference of fuzzy rules. Antecedent and consequent parameters of fuzzy rules are automatically defined by means of neuro-fuzzy learning. The system was tested on real-world meeting scenarios involving a different number of participants. The obtained results in terms of situation detection and responsiveness show that the proposed scheme can be successfully applied to recognize situations in any scenario regardless of the number of participants involved in the collaboration.

As a future work, we would like to perform a more extensive validation of our system by considering not only meetings, but also other social events, such as conferences, festivals and entertainment, which involve a larger number of users with different interaction patterns. In addition, we would like to integrate the CSA system with context aware services. For this reason, the architecture of the system has been designed to be easily integrated with different applications.

References

[1] N. Weißenberg, R. Gartmann, A. Voisard, An ontology-based approach to personalized situation-aware mobile service supply, Geoinformatica 10 (1) (2006) 55–90.
[2] C. Anagnostopoulos, S. Hadjiefthymiades, Enhancing situation-aware systems through imprecise reasoning, IEEE Trans. Mob. Comput. 7 (10) (2008) 1153–1168.
[3] J. Coutaz, J.L. Crowley, S. Dobson, D. Garlan, Context is key, Commun. ACM 48 (3) (2005) 49–53.
[4] A.K. Dey, Understanding and using context, Personal Ubiquitous Comput. 5 (1) (2001) 4–7.
[5] D. Vernon, G. Metta, G. Sandini, A survey of artificial cognitive systems: implications for the autonomous development of mental capabilities in computational agents, Trans. Evol. Comput. 11 (2) (2007) 151–180.
[6] A. Ciaramella, M. Cimino, B. Lazzerini, F. Marcelloni, Using context history to personalize a resource recommender via a genetic algorithm, in: 2010 10th International Conference on Intelligent Systems Design and Applications (ISDA), 2010, pp. 965–970.
[7] A. Ciaramella, M.G.C.A. Cimino, B. Lazzerini, F. Marcelloni, A situation-aware resource recommender based on fuzzy and semantic web rules, Int. J. Uncertain. Fuzziness Knowl. Based Syst. 18 (4) (2010) 411–430.
[8] P. Korpipää, J. Mantyjarvi, J. Kela, H. Keranen, E. Malm, Managing context information in mobile devices, IEEE Pervasive Comput. 2 (3) (2003) 42–51.
[9] J. Mäntyjärvi, T. Seppänen, Adapting applications in handheld devices using fuzzy context information, Interact. Comput. 15 (4) (2003) 521–538.
[10] A. Ranganathan, J. Al-Muhtadi, R.H. Campbell, Reasoning about uncertain contexts in pervasive computing environments, IEEE Pervasive Comput. 3 (2) (2004) 62–70.
[11] T. Gu, H.K. Pung, D.Q. Zhang, H.K. Pung, D.Q. Zhang, A Bayesian approach for dealing with uncertain contexts, in: 2004 Second International Conference on Pervasive Computing (Pervasive'04). Pervasive in the book, Advances in Pervasive Computing, Austrian Computer Society, Vienna, ustria, 2004.
[12] P. Delir Haghighi, S. Krishnaswamy, A. Zaslavsky, M.M. Gaber, Reasoning about context in uncertain pervasive computing environments, in: 3rd European Conference on Smart Sensing and Context, EuroSSC '08, Springer-Verlag, Berlin, Heidelberg, 2008, pp. 112–125.
[13] C. Anagnostopoulos, S. Hadjiefthymiades, Advanced fuzzy inference engines in situation aware computing, Fuzzy Sets Syst. 161 (4) (2010) 498–521.
[14] M.G. Cimino, B. Lazzerini, F. Marcelloni, A. Ciaramella, An adaptive rule-based approach for managing situation-awareness, Expert Syst. Appl. 39 (12) (2012) 10796–10811.
[15] M. Cimino, B. Lazzerini, F. Marcelloni, G. Castellano, A. Fanelli, M. Torsello, A collaborative situation-aware scheme for mobile service recommendation, in: 2011 11th International Conference on Intelligent Systems Design and Applications (ISDA), 2011, pp. 130–135.
[16] G. Castellano, M. Cimino, A. Fanelli, B. Lazzerini, F. Marcelloni, M. Torsello, A collaborative situation-aware scheme based on an emergent paradigm for mobile resource recommenders, J. Ambient Intell. Humaniz. Comput. (2012) 1–17, http://dx.doi.org/10.1007/s12652-012-0126-y.
[17] P. Barron, Using stigmergy to build pervasive computing environments (Ph.D. thesis), University of Dublin, Trinity College, October 2005.
[18] A. Bargiela, W. Pedrycz, Granular Computing: An Introduction, Kluwer Academic Publishers, Boston, MA, USA, 2003.
[19] J.-S. Jang, Anfis: adaptive-network-based fuzzy inference system, IEEE Trans. Syst. Man Cybern. 23 (3) (1993) 665–685.
[20] W. Pedrycz, Granular computing in multi-agent systems, in: Proceedings of the 3rd International Conference on Rough Sets and Knowledge Technology, RSKT'08, Springer-Verlag, Berlin, Heidelberg, 2008, pp. 3–17.
[21] M.G. Cimino, F. Marcelloni, Autonomic tracing of production processes with mobile and agent-based computing, Inf. Sci. 181 (5) (2011) 935–953.
[22] D. Greenwood, M. Lyell, A. Mallya, H. Suguri, The ieee fipa approach to integrating software agents and web services, in: Proceedings of the 6th International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS '07, ACM, New York, NY, USA, 2007, pp. 276:1–276:7.
[23] X. Wang, Z.M. Ma, L. Yan, X. Meng, Vague-swrl: a fuzzy extension of swrl, in: 2nd International Conference on Web Reasoning and Rule Systems (RR08), 2008, pp. 232–233.

[24] G. Sun, J. Chen, W. Guo, K.J.R. Liu, Signal processing techniques in network-aided positioning: a survey of state-of-the-art positioning designs, IEEE Signal Process. Mag. 22 (2005) 12–23.
[25] I. Horrocks, P.F. Patel-Schneider, A proposal for an owl rules language, in: Proceedings of the 13th International Conference on World Wide Web, WWW ′04, ACM, New York, NY, USA, 2004, pp. 723–731.
[26] G. Acampora, V. Loia, Using fml and fuzzy technology in adaptive ambient intelligence environments, Int. J. Comput. Intell. Res. 1 (2) (2005) 171–182.
[27] S. Bechhofer, F. van Harmelen, J. Hendler, I. Horrocks, D. McGuinness, P. Patel-Schneider, L. Stein, Owl Web Ontology Language Reference, W3c Recommendation, W3C, February 2004. URL ⟨http://www.w3.org/TR/owl-ref/⟩.
[28] A. Ciaramella, M.G.C.A. Cimino, F. Marcelloni, U. Straccia, Combining fuzzy logic and semantic web to enable situation-awareness in service recommendation, in: Proceedings of the 21st International Conference on Database and Expert Systems Applications: Part I, DEXA′10, Springer-Verlag, Berlin, Heidelberg, 2010, pp. 31–45.
[29] G. Acampora, M. Gaeta, V. Loia, A.V. Vasilakos, Interoperable and adaptive fuzzy services for ambient intelligence applications, ACM Trans. Auton. Adapt. Syst. 5 (2) (2010) 8:1–8:26.

Giovanna Castellano received the Laurea degree in Computer Science from the University of Bari, Italy, in 1993. From 1993 to 1995 she was a fellow researcher at the Institute for Signal and Image Processing (CNRBari). In 2001 she received the Ph.D. in Computer Science from the Department of Computer Science at the University of Bari, Italy, where she became Assistant Professor in 2002. Her research interests fall in the area of Computational Intelligence, with special focus on artificial neural networks, fuzzy systems, neuro-fuzzy modeling, granular computing, web personalization and fuzzy image processing. Within these research areas, she has been author of more than 160 papers published on scientific journals, books and conference proceedings. Currently, she serves as associate editor of Information Sciences and she is on the Editorial Board of two international journals.

Mario G.C.A. Cimino received the Ph.D. degree in Information Engineering from the University of Pisa, Italy, in 2007. He was a six-months visiting Ph.D. student at the Electrical and Computer Engineering Research Facility of the University of Alberta, Canada. He co-organized three editions of the Workshop on Computational Intelligence for Personalization in Web Content and Service Delivery. From 2008 to 2012, he was a Postdoctoral Research Fellow at the Department of Information Engineering of the University of Pisa, where he is currently a Researcher in Information Systems since June 2012. He is (co-) author of about 30 publications.

Anna Maria Fanelli is a Full Professor of Computer Science at the Department of Computer Science of the University of Bari, Italy. From 2005 to 2009 she was a Director of the Ph.D. School in Computer Science at the University of Bari. Currently, she is a Director of the Computer Science Department at the University of Bari and a chair of the CILab (Computational Intelligence Laboratory) at the same department. Her recent research interest focuses on the analysis, synthesis and applications of Computational Intelligence techniques with emphasis on the interpretability of fuzzy rule-based classifiers and Web Intelligence. She is co-author of more than 200 papers in international journals, books and conference proceedings. She is co-editor of international books on the above topics.

Beatrice Lazzerini is a Full Professor at the Department of Information Engineering of the University of Pisa, Italy, where she teaches “Computational Intelligence”, “Intelligent Systems” and “Decision Support Intelligent Systems”. Her research focus lies in the area of Computational Intelligence, with particular emphasis on fuzzy systems, neural networks and evolutionary computation. She has co-authored seven books and has published over 180 papers in international journals and conferences. She was involved in several national and international research projects. From 2003 to 2010 she was a President of the Specialized Laurea Degree in Computer Engineering for Enterprise Management of the University of Pisa.

Francesco Marcelloni received the Laurea degree in Electronics Engineering and the Ph.D. degree in Computer Engineering from the University of Pisa in 1991 and 1996, respectively. He is currently an Associate Professor at the University of Pisa. His main research interests include situation-aware service recommenders, multi-objective evolutionary fuzzy systems, energy-efficient data compression and aggregation in wireless sensor nodes, and monitoring systems for energy efficiency in buildings. He has co-edited three volumes, three journal special issues, and is (co-)author of a book and of more than 170 papers in international journals, books and conference proceedings. Currently, he serves as an associate editor of Information Sciences and is on the Editorial Board of other international journals.

Maria Alessandra Torsello received in 2008 the Ph.D. degree in Computer Science from the Department of Computer Science at the University of Bari, Italy, defending a thesis titled “Web Intelligence: a Neuro-Fuzzy Web Personalization System”. Her activity research mainly concerns the field of Computational Intelligence and the design of self-adaptive learning systems with particular interest in artificial neural networks, fuzzy logic, neuro-fuzzy modeling, knowledge-based neurocomputing, Computational Web Intelligence. She participated as a speaker to several International conferences. She co-organized three editions of the Workshop on Computational Intelligence for Personalization in Web Content and Service Delivery. She is co-author of about 40 publications.